COS 487: Theory of Computation **Fall 2000**

Assignment #3

Due, Monday, November , *Sanjeev Arora*

Suggested reading (for lectures 7,8,9,10,11): Sipser Chapters 3, 4, 5. Section 5.2 was not covered in class but should be perused.

Problems (from lectures 7, 8, 9, 10,11):

1. Give the transition diagram of a Turing Machine that accepts $\left\{ww^R : w \in \{0,1\}^*\right\}$.

2. Let the set of languages over the alphabet $\{0,1\}$ be divided into the following classes. *Class , ,* Regular. *Class , ,* Context-free but not Regular. *Class , ,* Decidable but not Context-free. *Class , ,* Recognizable but not Decidable. *Class , ,* Not Recognizable.

   If x is an integer, then let $[x]_2$ denote its binary representation. Classify each of the following languages and give a brief (1-2 lines) reason.

   (i) $\{[n]_2 : n \geq 0\}$.
   (ii) $\{[2^n]_2 : n \geq 0\}$.
   (iii) $\{[2^p]_2 : p$ is a prime number$\}$.
   (iv) $\{[2^{2n} + 2^n]_2 : n \geq 0\}$.
   (v) $\{[n]_2 :$ M accepts some string of length $\geq n\}$, where M is a Turing Machine.

3. Is the following language decidable? Prove your answer.

   $$\{<G>: G \text{ is a CFG that generates some string of even length}\}$$

4. Give short proofs of the following facts about mapping reducibility ($\leq_m$). For any languages A, B, C: (a) $A \leq_m A$. (b) If $A \leq_m B$ and $B \leq_m C$ then $A \leq_m C$. (c) If $A \leq_m B$ then $\overline{A} \leq_m \overline{B}$. (d) If A is enumerable and $A \leq_m \overline{A}$ then A is decidable. (e) If A is decidable then $A \leq_m 0^*1^*$. (f) If A is enumerable then $A \leq_m K_{TM}$

5. If u, v are binary strings, we say that $u < v$ if $1u$ represents a smaller integer than $1v$. Show that a language L is decidable iff the strings in L can be enumerated in an increasing order.

6. Let B be an infinite Turing-recognizable language consisting of descriptions of Turing machines $\{<M_1>, <M_2>, <M_3>, \ldots\}$. Show that there is a decidable language C consisting of Turing machines such that every machine in C has an equivalent Turing machine in B and vice versa. (Two Turing machines are *equivalent* if they have the same output on every input.)

7. In this question we explore the notion of *oracle reducibility.* If A is a language, then a *Turing machine with oracle A* is a machine with a "magical" subroutine that can decide membership in A. In other words, the subroutine, when given a string w, tells the machine whether or not $w \in A$. Show that there is a Turing machine with oracle $HALT_{TM}$ that can solve the following problem with *two* questions to its oracle: Given any three (machine, input) pairs $(<M_1, x_1>, <M_2, x_2>, <M_3, x_3>)$, decide for each pair whether or not it is in $HALT_{TM}$.

8. (a) Show that $\Re$ the set of real numbers, has the same size as $2^{\mathbf{N}}$, the power-set of the set of natural numbers. (ii) Show that $\Re \times \Re$, the set of all pairs of real numbers, has the same size as $\Re$. (*Aside, Can you think of a set whose size lies in between those of* $\mathbf{N}$ *and* $2^{\mathbf{N}}$*, This question has stumped mathematicians for many decades,*)